








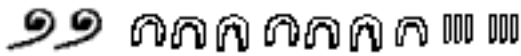
## 11787 Numeral Hieroglyphs

Egyptians had a writing system based on hieroglyphs from around 3000 BC. Hieroglyphs are small drawings representing words. Besides, Egyptians had a base 10 system of hieroglyphs for numerals. That is, they had separate symbols for one unit (a bar), ten units (an inverted 'U'), one hundred (a spiral), one thousand (a paper plant), ten thousand (a finger), one hundred thousand (a tadpole) and one million (a man kneeling).

These are the **numeral hieroglyphs**:

						
1	10	100	1000	10000	100000	10 <sup>6</sup>
Egyptian numeral hieroglyphs						

To make up number 276, for example, fifteen symbols were required: two “hundred” symbols, seven “ten” symbols, and six “unit” symbols. Number 276 would appear as:










Number 4622 would be represented as:



As you can see, Egyptians wrote ordered symbols, according to its value, from left to right as well as from right to left.

You have to convert numeral hieroglyphs into numbers. For that, we will use the following code:

						
B	U	S	P	F	T	M

So, we could represent 276 as SSUUUUUUUBBBBBB or BBBBBBUUUUUUSS. You cannot write more than nine times each character.

## Input

The first line of the input contains an integer,  $n$ , indicating the number of test cases. For each test case, one line appears, that contains a combination of  $m$  characters belonging to the following set {‘B’, ‘U’, ‘S’, ‘P’, ‘F’, ‘T’, ‘M’}, where  $1 \leq m \leq 500$ , representing, or not, possible numeral hieroglyphs.

## Output

For each combination of characters you must write either the corresponding number or the word ‘error’ if one of the two following cases occurs: (a) the input is not ordered, or (b) there are more than nine equal characters.

## Sample Input

```
8
PPPSUB
BUSPPP
PPPUJUPPP
BUSPFTM
MMMMMMMMM
MMMMTTTUBBBBB
BBPPPPPPFTTT
MMMMMMMMTTTTTTTTTTTTFFFFFFFFFFPPPPPPPPSSSSSSSSUUUUUUUUBBBBBBBB
```

## Sample Output

```
3111
3111
error
1111111
error
4300015
317002
9999999
```